

Google Cloud Platform

Using the instructions below one can establish an account on the Google Cloud. Once established, you will be able to perform whatever exercise is called for this semester

Signing up for the Google Cloud Platform

To sign up for the Free Trial, with a \$300 credit, you need a credit or debit card. Unfortunately, an American Express or other pre-paid Gift card will not work with Google Cloud.

To sign up go to:

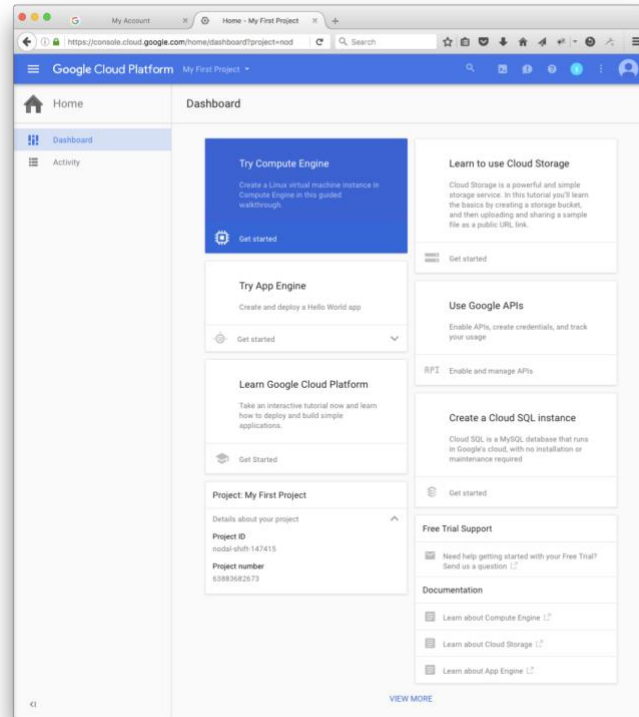
<https://console.cloud.google.com/freetrial?pli=1&page=0>

On the webpage entitled “Try Cloud Platform for free”, select that you “agree to use the services” and click on **Agree and continue**.

Figure 1: Google Cloud SignUp Screen

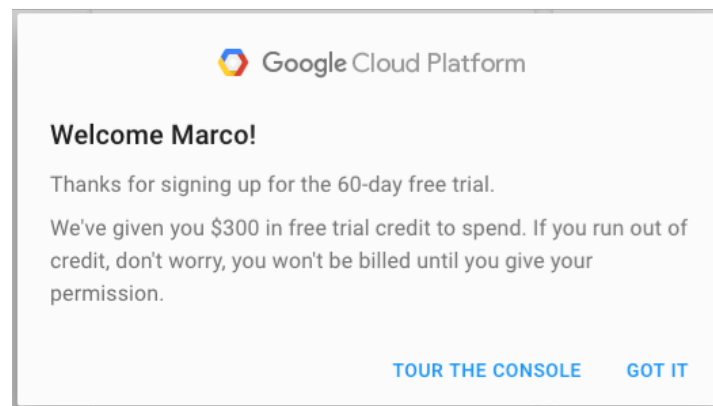
Select **Account Type Individual**. Follow the instructions to enter your account data as in **Figure 1**. You should **not** be using your @usc.edu e-mail account for your primary contact e-mail address, but instead use your @gmail.com address and finish by clicking **Start my free trial**.

Again, you will have to provide a credit or debit card.



After you are signed up, you will see the message “Creating project. This may take a few moments.” You will then be redirected to the **Dashboard** of the **Google Developer Console**.

A message will pop up indicating you signed up for the 60-day free trial, and have been given a \$300 free trial credit.



If you previously developed any projects using Google APIs, you will find them listed. You are now ready to proceed with the specifics of your homework/project.

Notes on Google Cloud Pricing

If you go to

<https://cloud.google.com/free/docs/always-free-usage-limits>

you will see that there are certain usage items that are always free. For example, under the Google App Engine 28 instance hours/day and 5GB Cloud Storage are just two of several items. The Google Cloud Datastore offers 1GB storage and 50,000 reads, 20,000 writes and 20,000 deletes for free. There are many other aspects of the Google Cloud that include free elements. Unfortunately, the DataProc is not one of them. As a result, the \$300 free credit will kick in immediately.

The Google Cloud Dataproc is more fully described at

<https://cloud.google.com/dataproc/>

Adopting Google Cloud Platform pricing principles, Cloud Dataproc has a low cost and an easy to understand **price structure, based on actual use, measured by the minute.**

Here is the structure:

CLOUD DATAPROC PRICING

Cloud Dataproc incurs a small incremental fee per virtual CPU in the Compute Engine instances used in your cluster¹.

Iowa	
MACHINE TYPE	PRICE
Standard Machines 1-64 Virtual CPUs	\$0.010 - \$0.640
High Memory Machines 2-64 Virtual CPUs	\$0.020 - \$0.640
High CPU Machines 2-64 Virtual CPUs	\$0.020 - \$0.640
Custom Machines Based on vCPU and memory usage	\$0.010/ vCPU hour

If you pay in a currency other than USD, the prices listed in your currency on [Cloud Platform SKUs](#) apply.

¹ Google Cloud Dataproc incurs a small incremental fee per virtual CPU in the Compute Engine instances used in your cluster while the cluster is operational. Additional resources used by Cloud Dataproc, such as a Compute Engine network, BigQuery, Cloud Bigtable, and others, are billed as they are consumed. For detailed pricing information, please view the [pricing guide](#).

Notice that Compute Engine, BigQuery, BigTable and “others” are billed.

Setting up Your Initial Machine

On the Dashboard, click on “Project” at the top of the window and either create a new project or select an existing one. For new projects choose a name. It may take a while to complete, but eventually you will be redirected to the Google cloud Dashboard.

Google has a large set of APIs, that will appear if you click on the menu immediately to the left of Google Cloud Platform. You will get a list that looks like **Figure 2** below. Included in the BIG DATA category are: BigQuery, Pub/Sub, Dataproc, Dataflow, Machine Learning and Genomics. For this exercise we will use Dataproc. Using Dataproc we can quickly create a cluster of compute instances running Hadoop. The alternative to Dataproc would be to individually setup each compute node, install Hadoop on it, set up HDFS, set up master node, etc. Dataproc automates this grueling process for us. Follow the instructions below to create a Hadoop cluster using Dataproc.

Creating a Hadoop Cluster on Google Cloud Platform

1. Create a Google Dataproc Cluster. Select **Dataproc** from the navigation list on the left

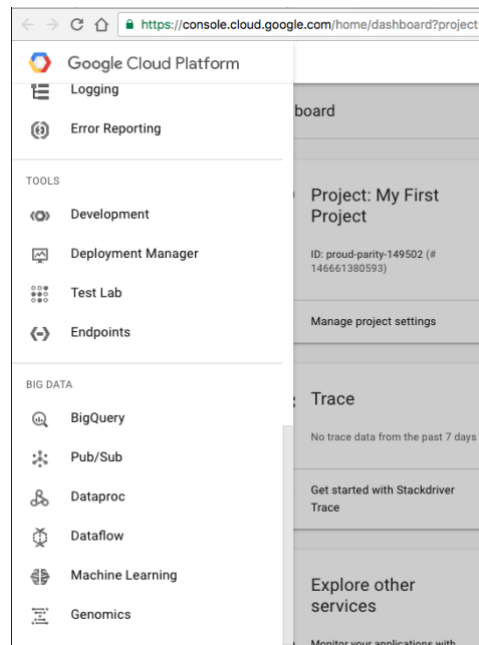


Figure 2: Google Cloud Platform APIs

2. If this is the first time you're using Dataproc then you'll encounter the error in the below screenshot (**Figure 3**). This means that your Google cloud account doesn't have the required API enabled. To enable the API copy the link in the error description and go to it. You will land on a page similar to the one in **Figure 4**. Click the **Enable** button at the top of the page to enable the Dataproc API.

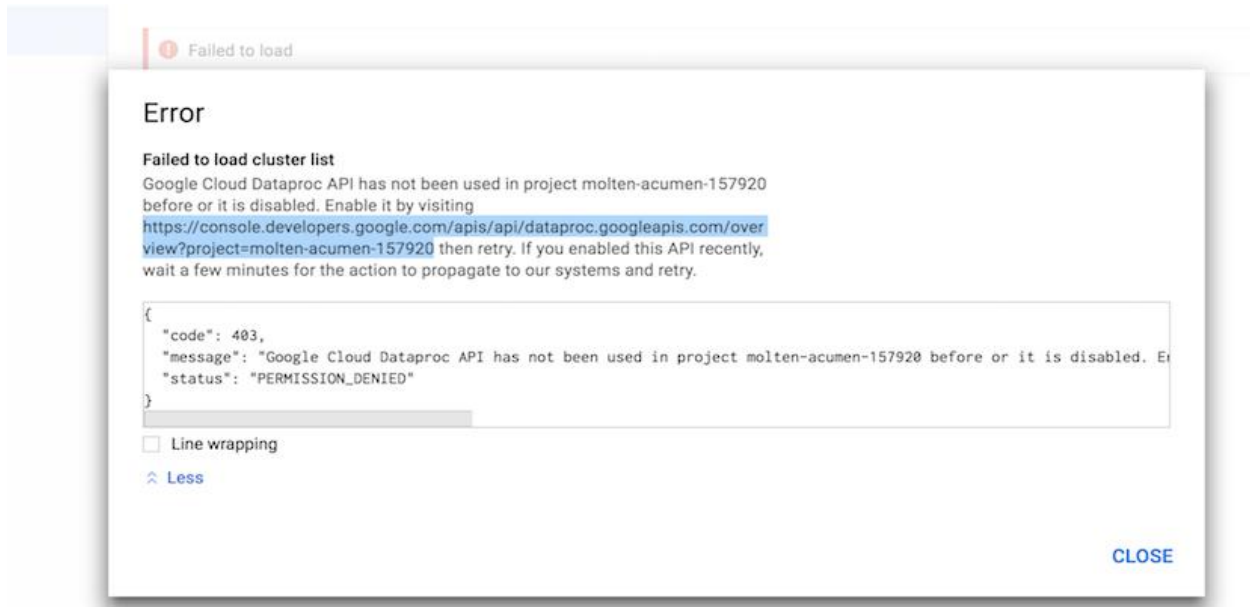


Figure 3: Error caused when trying to create a cluster for the first time

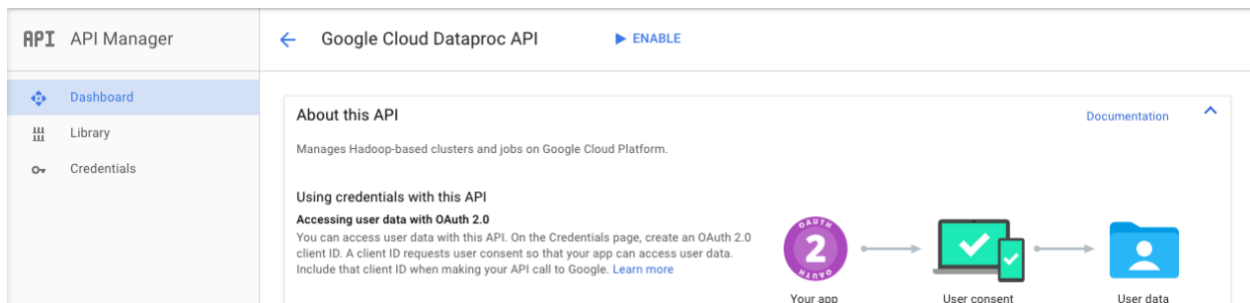


Figure 4: Enabling the Dataproc API

3. Once you've enabled the API go back to the page where you got the error earlier and reload the page. You'll now see a dialog box with a **Create Cluster** button (**Figure 5**).

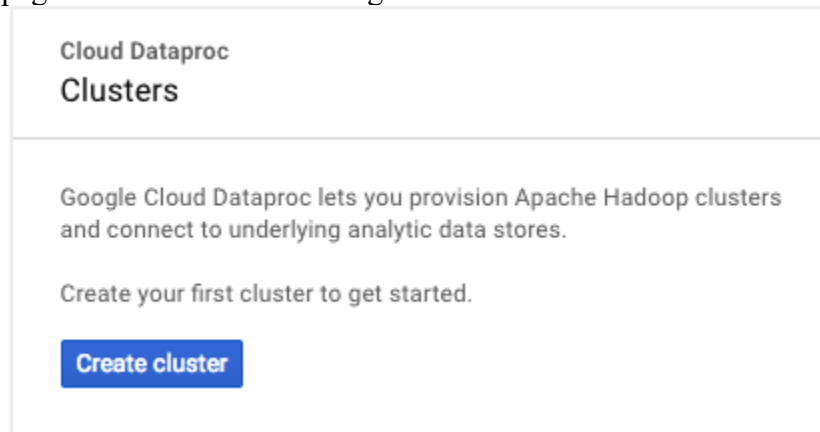


Figure 5: This is what you see once the API is enabled

4. Clicking on “**Create Cluster**” will take you to the cluster configuration section (**Figure 7**). Give any unique name to your cluster and select a **us-west** zone. You need to create a master and 3 worker nodes. Select the default configuration processors (**n1-standard-4**

4vCPU 15 GB memory) for each member and reduce the storage to **32 GB** HDD storage. Leave everything else default and click on “**Create**”. If you get an error (**Figure 6**) saying that you’ve exceeded your quota, reduce the number of worker nodes or choose a Machine Type(for master and worker) with fewer **vCPUs**. In rare cases you may get the error in **Figure 3** again. If so, simply follow the instructions in step 2 again. If all goes well your cluster will be created in a few minutes.

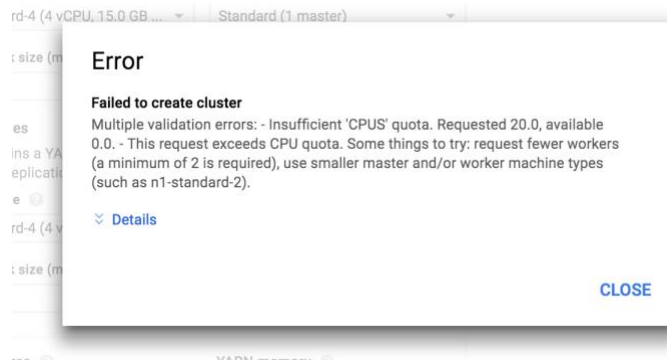


Figure 6: Insufficient CPU Quota error

Figure 7: Screen for setting up a cluster

5. Now that the cluster is setup we’ll have to configure it a little before we can run jobs on it. Select the cluster you just created from the list of clusters under the cloud Dataproc section on your console. Go to the **VM Instances** tab and click on the **SSH** button next to the instance with the **Master** Role. If you don’t see the SSH button click the **Refresh** button on the top of the page.

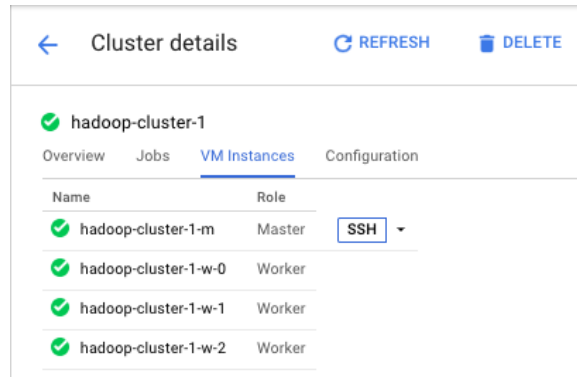


Figure 8: SSH into the master node.

6. Clicking on the **SSH** button will take you to a Command line Interface(CLI) like an xTerm or Terminal. All the commands in the following steps are to be entered in the CLI.

There is no home directory on HDFS for the current user so set up the user directory on HDFS. So, we'll have to set this up before proceeding further. (To find out your user name run whoami)



- `hadoop fs -mkdir -p /user/<your username here>`
7. Set up environment variables for JAVA and HADOOP_CLASSPATH. Please note that this step has to be done each time you open a new SSH terminal.
 - JAVA_HOME is already set-up. Do not change this.
 - `export PATH=${JAVA_HOME}/bin:${PATH}`
 - `export HADOOP_CLASSPATH=${JAVA_HOME}/lib/tools.jar`

To ensure that the environment variables are set, run the command `env`. You should see the path associated with JAVA_HOME in the PATH variable and a new variable called HADOOP_CLASSPATH as highlighted in the image below.

```
SSH_AUTH_SOCK=/tmp/ssh-7A1Rga0BUK/agent.7777
DAPPROC_MASTER_COMPONENTS=hadoop-hdfs-namenode hadoop-yarn-resourcemanager mysql-server
MAIL=/var/mail/adasari
PATH=/usr/lib/jvm/java-8-openjdk-amd64/bin:/usr/lib/jvm/java-8-openjdk-amd64/bin:/usr/lib/jvm/java-8-openjdk-amd64/bin:/usr/local/bin:/usr/bin:/bin:/usr/local/games:/usr/games
PWD=/home/adasari
JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
HADOOP_CLASSPATH=/usr/lib/jvm/java-8-openjdk-amd64/lib/tools.jar
LANG=en_US.UTF-8
DAPPROC_COMMON_COMPONENTS=openjdk-8-jdk libansi-java python-numpy libmvsq1-java hadoop-client hive pig spark-core spark-
```

8. Run `hadoop fs -ls`
9. If there is no error this implies that your cluster was successfully set up. If you do encounter an error it's most likely due to a missing environment variable or user home directory not being set up right. Retrace steps 1 to 6 to fix this.

NOTE:

- Please **disable** the billing for the cluster when you are not using it. Leaving it running will cost extra credits. The cluster is billed based on how many hours it is running and not how much data it is processing. So, if you leave the billing enabled overnight on an idle cluster you will still incur significant charges.
- Click the  on the top left corner in the Google console and go to the **Billing** section. Click the  button next to the project you created initially, and select *disable billing*. Please do this whenever you are not working on the cluster.
- See the “**Enable and Disable Billing account**” section on **page 11** for detailed instructions on how to do this.